STAT 4410/8416 Homework 5

ISLAM MD TAHIDUL

Due on Nov 28, 2023

1. **Working with databases:** Please follow the instruction below before answering the questions:

- Install the package `sqldf` using `install.packages('sqldf')`
- Import the library using `library('sqldf')`
- Read the file `https://raw.githubusercontent.com/dsindy/kaggle-titanic/master/data/train.csv` and store it in an object called `titanic`

We can now start writing SQL Script using `SQLDF library` right inside R. See example below:

```
library(sqldf)

sqldf("SELECT passengerid, name, sex
      FROM titanic
      limit 5", drv="SQLite")
```

```
##   PassengerId                                                 Name    Sex
## 1           1                              Braund, Mr. Owen Harris   male
## 2           2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female
## 3           3                               Heikkinen, Miss. Laina female
## 4           4       Futrelle, Mrs. Jacques Heath (Lily May Peel) female
## 5           5                             Allen, Mr. William Henry   male
```

Answer the following questions. Write SQL Script where applicable.

a. What does the following command do in MySQL?
   i. `show databases;` Answer: `show databases;` is used to display a list of all databases available on the MySQL server.

   ii. `show tables;` Answer: `show tables;` is used to display a list of all tables in the currently selected database.

b. Write `SQL` script to answer the following questions based on titanic data. Display the results of your script.
   i. What is the average age of passengers who survived? Group the data by Sex. Display only the column `Sex`, `AverageAge`

```
avg_age_passanger <- sqldf("SELECT Sex, AVG(Age) AS AverageAge
                           FROM titanic
                           WHERE Survived = 1 AND Age IS NOT NULL
                           GROUP BY Sex")
avg_age_passanger
```

```
##      Sex AverageAge
## 1 female   28.84772
## 2   male   27.27602
```

ii. What is the percentage of passengers who survived in each Passenger Class or `Pclass`? Group the data by `Sex`. Display Pclass, Sex, percentage value.

```
percent_passanger_servived <- sqldf("SELECT Pclass, Sex,
                                    (SUM(Survived) * 100.0 / COUNT(*)) AS
SurvivalPercentage
                                     FROM titanic
                                     GROUP BY Pclass, Sex")
percent_passanger_servived
```

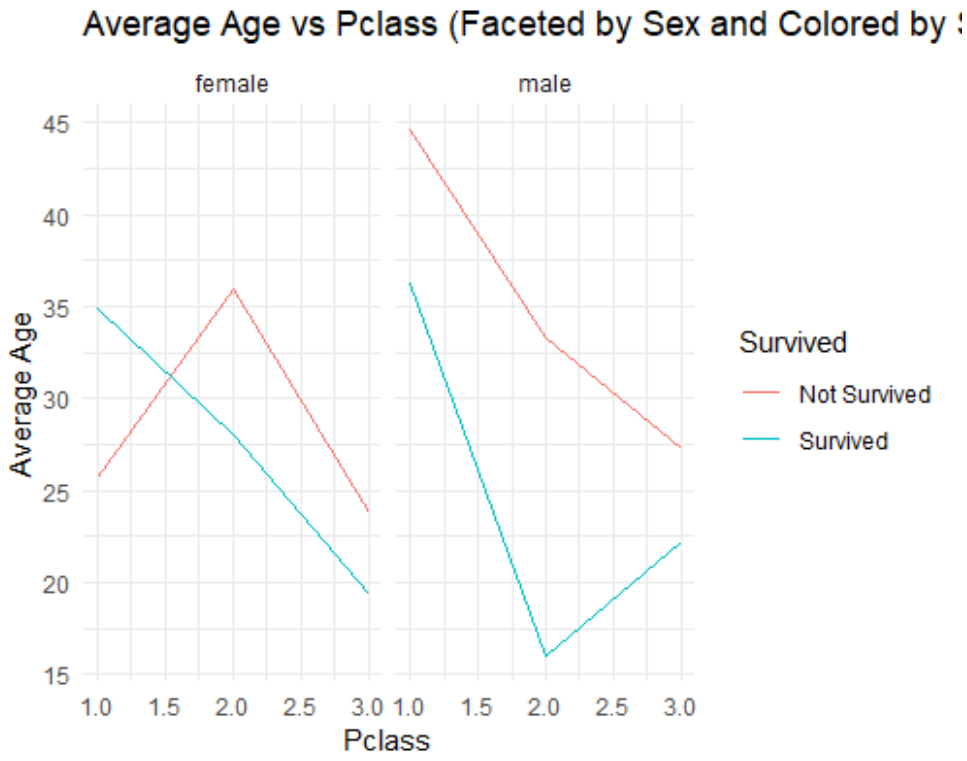
```
##   Pclass    Sex SurvivalPercentage
## 1      1 female           96.80851
## 2      1   male           36.88525
## 3      2 female           92.10526
## 4      2   male           15.74074
## 5      3 female           50.00000
## 6      3   male           13.54467
```

iii. What is the average age of all the passenger (survived and not survived)? Group the data by `Pclass`, `Sex`, `Survived`. After that use `ggplot` to generate a line plot to show average age vs pclass, facet by sex and color it by survived.

```
avg_age_ <- sqldf("SELECT AVG(Age) AS AverageAge
                   FROM titanic
                   WHERE Age IS NOT NULL")
avg_age_ <- round(avg_age_,2)
paste("average age of all the passenger (survived and not survived)
is",avg_age_)
```

```
## [1] "average age of all the passenger (survived and not survived) is 29.7"
```

```
avg_age <- sqldf("SELECT Pclass, Sex, Survived, AVG(Age) AS AvgAge
                  FROM titanic
                  WHERE Age IS NOT NULL
                  GROUP BY Pclass, Sex, Survived")
avg_age$Survived <- factor(avg_age$Survived, labels = c("Not Survived",
"Survived"))
library(ggplot2)
ggplot(avg_age, aes(x = Pclass, y = AvgAge, color = Survived)) +
  geom_line() +
  facet_grid(. ~ Sex) +
  labs(title = "Average Age vs Pclass (Faceted by Sex and Colored by
Survived)",
       x = "Pclass",
       y = "Average Age") +
  theme_minimal()
```



iv. What is the name, age, sex and pclass of the 5 oldest and 5 youngest persons who died?

```
oldest_died <- sqldf("SELECT Name, Age, Sex, Pclass
                     FROM titanic
                     WHERE Survived = 0 AND Age IS NOT NULL
                     ORDER BY Age DESC
                     LIMIT 5")
#print("5 oldest person who died is ")
oldest_died
```

```
##                          Name  Age  Sex Pclass
## 1         Svensson, Mr. Johan 74.0 male      3
## 2   Goldschmidt, Mr. George B 71.0 male      1
## 3     Artagaveytia, Mr. Ramon 71.0 male      1
## 4        Connors, Mr. Patrick 70.5 male      3
## 5 Mitchell, Mr. Henry Michael 70.0 male      2
```

```
youngest_died <- sqldf("SELECT Name, Age, Sex, Pclass
                       FROM titanic
                       WHERE Survived = 0 AND Age IS NOT NULL
                       ORDER BY Age
                       LIMIT 5")
youngest_died
```

```
##                                Name Age    Sex Pclass
## 1      Panula, Master. Eino Viljami   1   male      3
## 2   Goodwin, Master. Sidney Leonard   1   male      3
## 3    Palsson, Master. Gosta Leonard   2   male      3
```

```
## 4              Rice, Master. Eugene   2   male      3
## 5 Andersson, Miss. Ellis Anna Maria   2 female      3
```

v. On average which Passenger Class is more expensive?

```
avg_fare <- sqldf("SELECT Pclass, AVG(Fare) AS AverageFare
                   FROM titanic
                   GROUP BY Pclass
                   ORDER BY AverageFare DESC
                   LIMIT 1")
cat("The most expensive Passenger Class is ' Pclass:", avg_fare$Pclass[1], "'
and the fare is", avg_fare$AverageFare[1])
```

```
## The most expensive Passenger Class is ' Pclass: 1 ' and the fare is
84.15469
```

c. Notice the following R codes and explain what it is doing.

```
library(RSQLite)
conn <- dbConnect(RSQLite::SQLite(), "titanicDB")
dbWriteTable(conn, name = "titanic", value = titanic, overwrite=TRUE)
dbListTables(conn)
```

```
## [1] "titanic"
```

This R code establish a connection to an SQLite database, write the titanic data into a table named “titanic” in that database, and then list the tables in the connected database to verify the presence of the “titanic” table.

d. Use package `dplyr` to obtain the same result as you did in 1b(iii) above. For this use the connection string `conn` and the function `tbl()`. Store the result in an object called `meanAge`.

```
library(dplyr)
titanic_tbl <- tbl(conn, "titanic")
meanAge <- titanic_tbl %>%
  filter(!is.na(Age)) %>%
  group_by(Pclass, Sex, Survived) %>%
  summarise(AvgAge = mean(Age))
#meanAge
```

e. Show the SQL query to create `meanAge` in 1(d) using the function `show_query()`

```
show_query(meanAge)
```

```
## <SQL>
## SELECT `Pclass`, `Sex`, `Survived`, AVG(`Age`) AS `AvgAge`
## FROM (
##   SELECT `titanic`.*
##   FROM `titanic`
##   WHERE (NOT((`Age` IS NULL)))
## ) AS `q01`
## GROUP BY `Pclass`, `Sex`, `Survived`
```

2. **Working with MySQL:** Please follow the instruction below before answering the questions:

Obtain the remote access to the data science lab machine described in lecture 22 (UNO CAS Online Lab). This will allow you to work with MySQL from R. You may use RStudio in that virtual machine to work on this problem.

a. Use the following R code to connect the database trainingdb.

```
library(RMySQL)
con = dbConnect(MySQL(),user="training", password="training123",
                dbname="trainingDB", host="localhost")
```

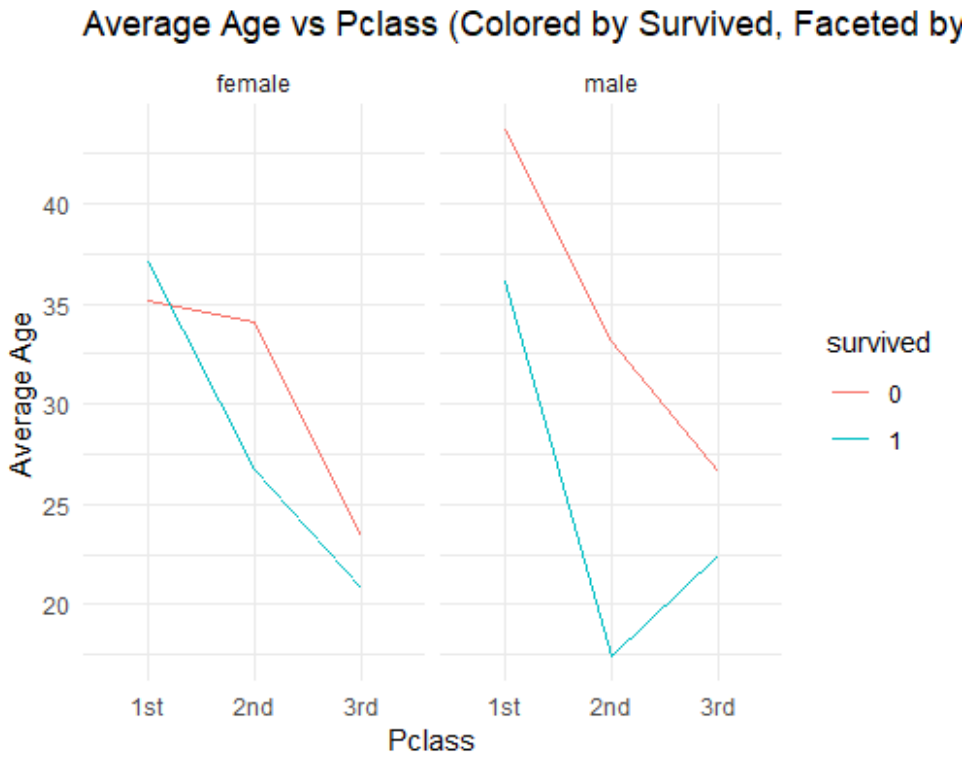
b. Write down a SQL command to select pclass, sex, survived and their average age from the titanic table. Store the selected data in data frame avgAge and display all the aggregated data.

```
#SELECT pclass, sex, survived, AVG(age) AS avg_age
#FROM titanic
#GROUP BY pclass, sex, survived;
avgAge <- dbGetQuery(con, "SELECT pclass, sex, survived, AVG(age) AS avg_age
FROM titanic GROUP BY pclass, sex, survived;")
avgAge
```

```
##    pclass    sex survived  avg_age
## 1     1st female        1 37.10938
## 2     1st   male        1 36.16824
## 3     1st female        0 35.20000
## 4     1st   male        0 43.65816
## 5     2nd   male        0 33.09259
## 6     2nd female        1 26.71105
## 7     2nd   male        1 17.44927
## 8     2nd female        0 34.09091
## 9     3rd   male        0 26.67960
## 10    3rd female        1 20.81482
## 11    3rd   male        1 22.43644
## 12    3rd female        0 23.41875
```

c. Now generate a line plot showing average age vs pclass colored by survived and faceted by sex.

```
library(ggplot2)
avgAge$pclass <- factor(avgAge$pclass, levels = c("1st", "2nd", "3rd"))
ggplot(avgAge, aes(x = pclass, y = avg_age, group = survived, color =
survived)) +
  geom_line() +
  facet_grid(. ~ sex) +
  labs(title = "Average Age vs Pclass (Colored by Survived, Faceted by Sex)",
       x = "Pclass",
       y = "Average Age") +
  theme_minimal()
```



d. Use the package `dplyr` to obtain the same result as you did in question 2(b). Display the results and the underlying `SQL` command used by `dplyr`.

```
avgAge_ <- titanic_tbl %>%
  group_by(Pclass, Sex, Survived) %>%
  summarise(AvgAge = mean(Age, na.rm = TRUE))

show_query(avgAge_)

## <SQL>
## SELECT `Pclass`, `Sex`, `Survived`, AVG(`Age`) AS `AvgAge`
## FROM `titanic`
## GROUP BY `Pclass`, `Sex`, `Survived`
```